IN THE CLAIMS:

Please amend claims 1-10 and add new claims 11-20 as follows:

1.   (Currently amended)   A method of converting digital data, the method comprising ~~the steps of~~:

binding input digital data into unit blocks ~~constituted by~~ comprising a plurality of bytes;

modulation-coding each byte of the ~~input data~~ unit blocks ~~by using~~ according to a code conversion table; and

allocating ~~a~~ at least ~~merging~~ one merging bit ~~in a block unit~~ for ~~the~~ each modulation-coded ~~input data in a block~~ unit block.

2.   (Currently amended)   The method of claim 1, wherein ~~the~~ each of the input data block~~s~~ ~~is bound in the unit of~~ comprises three to seven bytes.

3.   (Currently amended)   The method of claim 1, wherein ~~the~~ three merging bit~~s~~ are allocated ~~in a block unit is made of three bits~~.

4.   (Currently amended)   The method of claim 1, wherein each ~~of the~~ modulation-coded ~~input data~~ unit block is encoded into a code word of a fifteen bit~~s~~ ~~length~~ ~~by~~ according to an 8/15 conversion table.

5.   (Currently amended)   The method of claim 1, wherein a running digital sum (RDS) of ~~the latest input data~~ a present unit block is compared to ~~a~~ an RDS of ~~the~~ a previous ~~input data~~ unit block to ~~select~~ allocate the at least one merging bit ~~so~~ such that the RDS ~~has the minimum value~~ is minimized without violating run length limited (RLL) restraints ~~when the latest input data block is produced~~.

6.   (Currently amended)   The method of claim 5, wherein the ~~selected~~ at least one merging bit is primarily outputted, and the ~~encoded~~ modulation-coded present ~~input data~~ unit block is outputted, and ~~simultaneously~~ the running digital sum (RDS) up to the ~~current~~ present

unit block is <u>simultaneously</u> updated to prepare <u>for allocation of at least one</u> merging bit ~~selection of the~~<u>for a</u> next <u>unit</u> block.

7.   (Currently amended)   A method of digital data conversion, comprising ~~the steps of~~:

performing 8/15 modulation-coding ~~to~~<u>of</u> an input data block ~~in the unit of~~<u>of</u> m byte<u>s</u> and simultaneously producing a <u>running digital sum (RDS)</u> of the <u>input data</u> block ~~in the corresponding order~~;

evaluating the RDS of the <u>input data</u> block ~~in the corresponding order~~ and ~~the~~ <u>an</u> RDS of ~~the~~ <u>a</u> previous <u>input data</u> block to select ~~a~~ <u>at least one</u> merging bit; and

outputting the selected <u>at least one</u> merging bit, ~~following~~ <u>followed</u> by ~~outputting~~ the ~~encoded~~ <u>modulation-coded input data</u> block ~~in the corresponding order~~, and updating the RDS for selecting ~~a~~ <u>at least one</u> merging bit ~~in the~~<u>for a</u> next <u>input data</u> block ~~in order~~.

8.   (Currently amended)   A method of recording and reproducing digital data, <u>the method</u> comprising ~~the steps of~~:

binding input digital data into unit blocks ~~constituted~~ <u>comprising</u> ~~by~~ a plurality of bytes and modulation-coding <u>each of</u> the ~~bytes~~<u>unit blocks</u>;

allocating ~~a~~ <u>at least one</u> merging bit ~~in block unit for the~~<u>for each</u> modulation-coded ~~input data in the~~<u>unit</u> block ~~unit~~;

recording ~~a~~ byte-unit information ~~expressing~~ <u>indicating</u> the number of ~~the~~ bytes ~~constituting~~ <u>comprising</u> each ~~of the data~~<u>unit</u> blocks together with data added with the <u>at least</u> ~~merging~~<u>one merging</u> bit after modulation-coding; and

~~performing data~~ decoding ~~for~~ the corresponding <u>unit</u> block ~~by~~ using the recorded byte-unit information.

9.   (Currently amended)   The method of claim 8, wherein each of the ~~input data~~<u>unit</u> blocks ~~is~~<u>comprises</u> ~~modulation-coded and decoded in the unit of~~ three to seven bytes.

10.   (Currently amended) The method of claim 8, wherein the <u>at least one</u> merging bit is ~~selected from the merging bits of three bit so~~<u>allocated such</u> that ~~the~~<u>a</u> running digital sum

(RDS) value ~~may be minimum while not~~ is minimized without violating ~~the~~ run length limited (RLL) restraints.

11.   (New)   A method of converting digital data, the method comprising:

binding input digital data into unit blocks comprising a plurality of bytes;

modulation-coding each of the plurality of bytes of the unit blocks according to a code conversion table; and

comparing an RDS of a present unit block to an RDS of a previous unit block to allocate at least one merging bit for the present modulation-coded unit block such that the RDS is minimized without violating RLL restraints,

wherein the at least one merging bit is primarily outputted, and the modulation-coded present unit block is outputted, and the RDS up to the present unit block is simultaneously updated to prepare for allocation of at least one merging bit for a next unit block.

12.   (New)   The method of claim 11, wherein each unit block comprises three to seven bytes.

13.   (New)   The method of claim 11, wherein three merging bits are allocated for each modulation-coded unit block.

14.   (New)   The method of claim 11, wherein each unit block is modulation-coded into a code word of a fifteen bits according to an 8/15 conversion table.

15.   (New)   The method of claim 7, wherein m is three to seven bytes.

16.   (New)   The method of claim 7, wherein three merging bits are selected.

17.   (New)   The method of claim 8, wherein three merging bits are allocated for each unit block.

18.   (New) The method of claim 8, wherein each of the unit blocks is modulation-coded into a code word of a fifteen bits according to an 8/15 conversion table.

19.   (New) The method of claim 10, further comprising comparing an RDS of a present unit block to an RDS of a previous unit block to allocate the at least one merging bit.

20.   (New) The method of claim 19, wherein the at least one merging bit is primarily outputted, and the modulation-coded present unit block is outputted, and the RDS up to the present unit block is simultaneously updated to prepare for allocation of at least one merging bit for a next unit block.

Attorney Docket No. 2080-3-44